Sharif University of Technology

Department of Computer Engineering

Object-Oriented Systems Design (40484)

Raman Ramsin

Undergraduate Course, 3 Units, Core Elective (BSc in Computer Engineering)
Prerequisite: Systems Analysis and Design (40418)

Overview

The aim of this course is to familiarize undergraduate students of Computer Engineering with the concepts, principles, and methods of object-oriented systems analysis and design. While gaining knowledge and hands-on experience with UML and a prominent third-generation object-oriented software development methodology, students will also be introduced to GoF design patterns and their practical use in software development projects.

Topics

1) Introduction – a review of object oriented concepts, and the evolution of object-oriented analysis and design methods (1 session – each session is 90 minutes in duration)
2) A review of the Unified Modeling Language – UML (4 sessions)
3) Phases and workflows (disciplines) in USDP
   a) The four phases of USDP (3 sessions)
   b) Requirements workflow – identification and specification of use cases (3 sessions)
   c) Analysis workflow
      i) Identification and modeling of analysis classes and objects (2 sessions)
      ii) Identification and modeling of relationships among analysis classes and objects (2 sessions)
      iii) Analysis packages (1 session)
      iv) Use case realizations as pertaining to analysis (2 sessions)
      v) Modeling of activities (2 sessions)
   d) Design workflow
      i) Identification and modeling of design classes and objects (1 session)
      ii) Refinement of class relationships (1 session)
      iii) Interfaces and components (1 session)
      iv) Use case realizations as pertaining to design (1 session)
   e) Implementation workflow (1 session)
4) Design patterns
   a) Principles and rules of object-oriented analysis and design: Basic principles, GRASP patterns, and Design by Contract (1 session)
   b) GoF design patterns
      i) Creational patterns: *Factory Method*, *Abstract Factory*, *Builder*, *Prototype*, and *Singleton* (1 session)
      ii) Structural patterns: *Adapter*, *Bridge*, *Composite*, *Decorator*, *Facade*, and *Proxy* (1 session)
      iii) Behavioral patterns: *Chain of Responsibility*, *Iterator*, *Mediator*, *Memento*, *Observer*, *State*, *Strategy*, and *Visitor* (2 sessions)

Assessment

- Two exams (Midterm and Final) – Comprising 60% of the total grade
- One comprehensive course project: Project activities will be assigned and completed throughout the semester – Comprising 40% of the total grade

Main References

- J. Arlow and I. Neustadt, *UML 2 and the Unified Process,* 2nd ed. Addison-Wesley, 2005.
- H. Gomaa, *Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures.* Cambridge University Press, 2011.
- G. Booch, R.A. Maksimchuk, M.W. Engel, B.J. Young, J. Conallen, and K.A. Houston, *Object-Oriented Analysis and Design with Applications,* 3rd ed. Addison-Wesley, 2007.
- E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software.* Addison-Wesley, 1995.
- C. Larman, *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*, 3rd ed. Prentice-Hall, 2004.